



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/672,514	09/25/2003	Thomas Gauweiler	11884/401203	1167
26646 7590 10/16/2007 KENYON & KENYON LLP ONE BROADWAY NEW YORK, NY 10004			EXAMINER TIMBLIN, ROBERT M	
			ART UNIT 2167	PAPER NUMBER
			MAIL DATE 10/16/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/672,514

Applicant(s)

GAUWEILER, THOMAS

Examiner

Robert M. Timblin

Art Unit

2167

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 08 August 2007.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1, 3 and 9-15 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1, 3, and 9-15 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

This office action corresponds to application 10/672,514 and Applicant's remarks/amendments filed 8/8/2007.

Response to Amendment

The amendments entered 8/8/2007 have been acknowledged and entered. Claims 1, 3, and 9-15 are pending in this application.

Claim Objections

Claim 1 is objected to because of the following informalities: step (d) should read when a next record requested *from* the first record set (i.e. step (a) recites retrieving from a recordset instead of *by* a first recordset). Claim 9 is objected to for the same reason.

Claim 9 is objected to because step (c) should include reading *steps* (a) and (b).

Claim 10 is objected to because the second determining step should end with a semicolon (;) instead of a (:).

Appropriate correction is required.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claim 1 and 9 recites the limitation "the first recordset and at least one second recordset" in step (e). There is insufficient antecedent basis for the second recordset in the claim.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1, 3, 9, 11, and 12 are rejected under 35 U.S.C. 103(a) as being obvious over Nguyen et al. (Nguyen hereafter) (U.S. Patent 6,216,137) in view of Kodavalla et al. (Kodavalla hereafter (U.S. Patent 5,717,919).

With respect to claim 1, Nguyen teaches A method for managing bufferpages and redundant copies of records in a local memory associated with a mobile device application, comprising:

(a) retrieving (figure 1b; data retrieval unit, and col. 6 line 38-45 teaches retrieving data) a first record (col. 6 line 48-53 and figures 3a-4a) from a remote database memory (figure 1b) in response to a request (abstract, i.e. an application requesting data and col. 6 line 58) from a first recordset (col. 6 line 48, i.e. a set of data);

(b) saving the first record (figure 4a, drawing reference 300) on a first bufferpage of the local memory (figure 1a, drawing reference 100) associated with the mobile device application (col. 4 line 33-39 and figure 1b drawing reference 180; i.e. a use of a stylus and touch screen

Art Unit: 2167

suggest a mobile device), the first bufferpage being associated with the first recordset (col. 6 line 48, i.e. a set of data);

(c) repeating steps (a) and (b) for at least one further record (figures 3a and 4a, i.e. retrieving newer data 302-306);

(e) determining (figure 1b, drawing reference 196, and col. 6 line 55-64) if one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) was previously retrieved (col. 3 line 7-8 and col. 7 line 34-35) and saved (drawing reference 514) in the local memory (figure 1a, drawing reference 100) associated with the mobile device application (col. 4 line 33-39 and figure 1b, drawing reference 180) by at least one of the first recordset and at least one second recordset (col. 6 line 48, i.e. a set of data) as one of a prior saved version (col. 3 line 7-8 and col. 7 line 34-35, and figures 3a-4a) of the first record, a prior saved version of the at least one further record, and a prior saved version of the next record, respectively (i.e. figures 3a-4a suggest storing records in accordance with their previously retrieved versions); and

(f) storing a pointer (figures 3a-4a show pointers pointing to prior versions of records) with one of the prior saved version of the first record (e.g. drawing reference 300), the prior saved version of the at least one further record (e.g. drawing reference 302), and the prior saved version of the next record (e.g. drawing reference 304), the pointer pointing to the one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) if one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) was previously retrieved and saved (col. 3 line 7-8 and col. 7 line 34-35)

as one of the prior saved version of the first record (e.g. drawing reference 300), the prior saved version of the at least one further record (e.g. drawing reference 302), and the prior saved version of the next record (e.g. drawing reference 304).

Nguyen fails to explicitly disclose the use of bufferpages for saving records. That is, Nguyen fails to teach saving the first record on a first bufferpage as well as the step (d) of when a next record requested by the first recordset is larger than a freespace on the first bufferpage, saving the next record on a second bufferpage of the local memory associated with the mobile device application, the second bufferpage being associated with the first recordset. Kodavalla, however, teaches saving the first record on a bufferpage as saving data records on a data page for holding records (Kodavalla at col. 7 line 29-42). Further, Kodavalla teaches the step (d) of when a next record requested by the first recordset is larger than a freespace on the first bufferpage, saving the next record on a second bufferpage of the local memory associated with the mobile device application, the second bufferpage being associated with the first recordset (col. 7 line 43-47) to allocate a new page to store data if insufficient room exists.

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because Nguyen could have used Kodavalla's teachings to efficiently manage memory for storing records in limited memory environments. Nguyen could have benefited from such a use as they describe utilizing limited memory (i.e. RAM, col. 4 line 5) in their system.

Nguyen also fails to explicitly teach otherwise creating a business object kernel pointing to one of the first record, the at least one further record and the next record.

Kodavalla, however, teaches otherwise creating a business object kernel pointing to one of the first record, the at least one further record and the next record (col. 4 line 35 and col. 1 line 60) to allocate memory for new data (e.g. Kodavalla teaches at col. 20, line 55 to allocate a first page).

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because Nguyen could have used the kernel taught by Kodavalla for efficiently allocating memory. Further, if the to one of the first record, the at least one further record and the next record have not been retrieved as prior versions, respectively the kernel of Kodavalla would have been effective in managing different versions of different data records (i.e. by allocating a pointer to the versions).

With respect to claim 3, Nguyen fails to expressly teach comparing the freespace on the first bufferpage to a size of the next record.

Kodavalla, however, teaches the method of claim 1, further comprising comparing the freespace on the first bufferpage to a size of the next record as maintaining a free list containing a list of free spaces with sufficient room for storing at he particular record being inserted (col. 2, lines 32-35). Furthermore a comparison is taught by determining if a page is full (col. 7, lines 29-35).

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because in use of Kodavalla's

teachings, Nguyen could have determined when to allocate more memory for new versions of data for the benefit of efficiently managing data in a limited memory environment.

With respect to claim 9, Nguyen teaches, A system for managing bufferpages and redundant copies of records of a mobile device application, comprising:

a remote database memory (figure 1b, drawing reference 188);

a local program memory (figure 1a, drawing reference 100) associated with the mobile device application (col. 4 line 33-39 and figure 1b drawing reference 180; i.e. a use of a stylus and touch screen suggest a mobile device); and

a local mobile processor (drawing reference 102) coupled to the remote database memory (figure 1b, drawing reference 188) and the local program memory (figure 1a, drawing reference 100) associated with the mobile device application (drawing reference 180), the local mobile processor (drawing reference 120) adapted to:

(a) retrieve (figure 1b; data retrieval unit, and col. 6 line 38-45 teaches retrieving data) a first record (col. 6 line 48-53 and figures 3a-4a) from a remote database memory (figure 1b) in response to a request (abstract, i.e. an application requesting data and col. 6 line 58) from a first recordset (col. 6 line 48, i.e. a set of data);

(b) save the first record on a first bufferpage of the local program memory associated with the mobile device application, the first bufferpage being associated with the first recordset;

(c) repeating steps (a) and (b) for at least one further record (figures 3a and 4a, i.e. retrieving newer data 302-306);

(e) determine (figure 1b, drawing reference 196, and col. 6 line 55-64,) if one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) was previously retrieved (col. 3 line 7-8 and col. 7 line 34-35) and saved (drawing reference 514) in the local memory (figure 1a, drawing reference 100) associated with the mobile device application (col. 4 line 33-39 and figure 1b drawing reference 180) by at least one of the first recordset and at least one second recordset (col. 6 line 48, i.e. a set of data) as one of a prior saved version (col. 3 line 7-8 and col. 7 line 34-35, and figures 3a-4a) of the first record, a prior saved version of the at least one further record, and a prior saved version of the next record, respectively (i.e. figures 3a-4a suggest storing records in accordance with their previously retrieved versions); and

(f) store a pointer (figures 3a-4a show pointers pointing to prior versions of records) with one of the prior saved version of the first record (e.g. drawing reference 300), the prior saved version of the at least one further record (e.g. drawing reference 302), and the prior saved version of the next record (e.g. drawing reference 304), the pointer pointing to the one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) if one of the first record (e.g. drawing reference 300), the at least one further record (e.g. drawing reference 302), and the next record (e.g. drawing reference 304) was previously retrieved and saved (col. 3 line 7-8 and col. 7 line 34-35) as one of the prior saved version of the first record (e.g. drawing reference 300), the prior saved version of the at least one further record (e.g. drawing reference 302), and the prior saved version of the next record (e.g. drawing reference 304), respectively, otherwise creating a business object kernel pointing to one of the first record, the at least one further record, and the next record.

Nguyen fails to explicitly disclose the use of bufferpages for saving records. That is, Nguyen fails to teach saving the first record on a first bufferpage as well as the step (d) of when a next record requested by the first recordset is larger than a freespace on the first bufferpage, saving the next record on a second bufferpage of the local memory associated with the mobile device application, the second bufferpage being associated with the first recordset. Kodavalla, however, teaches saving the first record on a bufferpage as saving data records on a data page for holding records (Kodavalla at col. 7 line 29-42). Further, Kodavalla teaches the step (d) of when a next record requested by the first recordset is larger than a freespace on the first bufferpage, saving the next record on a second bufferpage of the local memory associated with the mobile device application, the second bufferpage being associated with the first recordset (col. 7 line 43-47) to allocate a new page to store data if insufficient room exists.

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because Nguyen could have used Kodavalla's teachings to efficiently manage memory for storing records in limited memory environments. Nguyen could have benefited from such a use as they describe utilizing limited memory (i.e. RAM, col. 4 line 5) in their system.

Nguyen also fails to explicitly teach otherwise creating a business object kernel pointing to one of the first record, the at least one further record and the next record.

Kodavalla, however, teaches otherwise creating a business object kernel pointing to one of the first record, the at least one further record and the next record (col. 4 line 35 and col. 1 line

Art Unit: 2167

60) to allocate memory for new data (e.g. Kodavalla teaches at col. 20, line 55 to allocate a first page).

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because Nguyen could have used the kernel taught by Kodavalla for efficiently allocating memory. Further, if the to one of the first record, the at least one further record and the next record have not been retrieved as prior versions, respectively the kernel of Kodavalla would have been effective in managing different versions of different data records (i.e. by allocating a pointer to the versions).

With respect to claim 11, Nguyen teaches the method of claim 1, wherein determining further comprises checking a look-up table (drawing reference 316 and col. 9 line 65-67).

With respect to claim 12, Nguyen teaches the system of claim 9, wherein the local mobile processor is adapted to determine if one of the first record, the at least one further record, and the next record was previously retrieved and saved as the prior record by checking a look-up table (drawing reference 316 and col. 9 line 65-67).

Claims 10 and 13-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kodavalla in view of Nguyen as applied to claims 1, 3, 9, 11, and 12 above (i.e. Nguyen in view of Kodavalla).

With respect to claim 10, Kodavalla teaches A method of managing fixed units of buffer memory associated with a mobile client application, comprising:

retrieving a record stored in a remote database memory (as the clients issue a query for retrieving particular data meeting the query condition from table 250 col. 6, lines 14-24 and figure 2);

determining a size of the retrieved record and a size of a freespace of a current fixed unit of buffer memory (col. 7 lines 43-50) and

saving the retrieved record in the current fixed unit of buffer memory if the size of the retrieved record is smaller than the free space of the current fixed unit of buffer memory (col. 7 lines 43-45);

saving the retrieved record in a next fixed unit of buffer memory if the size of the retrieved record is larger than the freespace of the current fixed unit of buffer memory (col. 7 lines 30-40; adding a new page when full);

creating a business object kernel including a key pointing to the fixed unit of buffer memory storing the new copy of the retrieved record, if the retrieved record was not previously retrieved and stored (drawing reference 300, col. 7 lines 34-41, and figures 3A-B).

Kodavalla does not expressly teach determining if the retrieved record was previously retrieved and stored by the mobile client application. Kodavalla also fails to expressly teach storing a pointer pointing from a fixed unit of buffer memory storing a most recent copy of the retrieved record to a fixed unit of buffer memory storing a new copy of the retrieved record, if the retrieved record was previously retrieved and stored by the mobile client application.

Nguyen, however, teaches determining (figure 1b, drawing reference 196, and col. 6 line 55-64) if the retrieved record was previously retrieved and stored (col. 3 line 7-8 and col. 7 line 34-35) and saved (drawing reference 514) by the mobile client application (col. 4 line 33-39 and figure 1b, drawing reference 180). Nguyen also teaches storing a pointer (figures 3a-4a show pointers pointing to prior versions of records) pointing from a fixed unit of buffer memory storing a most recent copy of the retrieved record to a fixed unit of buffer memory storing a new copy of the retrieved record, if the retrieved record was previously retrieved and stored by the mobile client application (i.e. Nguyen teaches in figures 3a-4a of a new version (i.e. interpreted as a “copy”) pointing to a most recent version).

In the same field of endeavor, (i.e. controlling redundant data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because Kodavalla could have used Nguyen’s teachings to manage a plurality of copies and thus efficiently handle redundant data. Kodavalla shows a need for controlling redundant data (col. 6 line 57).

Further, Kodavalla discloses a concern for formatting data for insertion (i.e. at col. 26 line 26-27). Nguyen, who is also in the endeavor of formatting data, would have benefited Kodavalla as they provide converting data to an expected format (col. 2 line 57-60) for ease of inserting data.

With respect to claim 13, Kodavalla fails to expressly teach the method of claim 10, wherein determining if the retrieved record was previously retrieved and stored by the mobile client application comprises checking a look-up table.

Art Unit: 2167

Nguyen, however, teaches determining if the retrieved record was previously retrieved and stored by the mobile client application comprises checking a look-up table (drawing reference 316 and col. 9 line 65-67).

With respect to claim 14, Kodavalla teaches the method of claim 10, further comprising storing the business object kernel in a look-up table (col. 3 line 49-50, col. 7 line 43-45, and col. 10 line 5-15).

With respect to claim 15, Kodavalla fails to expressly teach wherein the key comprises a counter indicating a number of times the retrieved record is stored.

Nguyen, however, teaches wherein the key comprises a counter indicating a number of times the retrieved record is stored (figure 4a, i.e. VERSION_1-VERSION_4) for keeping track of how many versions of the same record are stored.

In the same field of endeavor, (i.e. forming chained data structures for storing data), it would have been obvious to one of ordinary skill in the data processing art at the time of the present invention to combine the teachings of the cited references because keeping track of the number of times a retrieved data is stored would have given Kodavalla a way to keep a count of allocations in a data chain (needed by Kodavalla at col. 56 line 12 of the SL_CHGVALUE function).

Response to Arguments

Applicant's arguments with respect to claims 1,3, and 9-15 have been considered but are moot in view of the new ground(s) of rejection. Applicant's amendments have necessitated new grounds of rejection and therefore the arguments are moot.

As presented above, Nguyen in view of Kodavalla teaches claims 1 and 9 as claimed. Claim 10 contains similar limitations thereto and is rejected respectively in the above rejection.

Furthermore, and in response to Applicant's argument on page 10 of the response, Nguyen teaches the use of pointers to link different versions of the same individual record as seen in the above rejection.

Applicant's arguments, in the response, filed 8/8/2007, with respect to claims 13 and 15 have been fully considered and are persuasive. The previous rejection of claims 13 and 15 have been withdrawn. However, in light of the new rejection in view of Nguyen, arguments to claims 13 and 15 are moot.

In response to applicant's arguments, the recitation the management of redundant copies of individual records has not been given patentable weight because the recitation occurs in the preamble. A preamble is generally not accorded any patentable weight where it merely recites the purpose of a process or the intended use of a structure, and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone. See *In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

Furthermore, and in response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies

Art Unit: 2167

(i.e., management of redundant copies of *individual* records) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure for being concerned with version management.

U.S. Patent 6,631,386 to Arun.

U.S. Patent 5,347,653 to Flynn.

U.S. Patent 5,794,229 to French.

U.S. Patent 5,440,730 to Elmasr.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert M. Timblin whose telephone number is 571-272-5627. The examiner can normally be reached on M-F 8:00-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John R. Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.


Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications

Art Unit: 2167

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Robert M. Timblin

Patent Examiner AU 2167



JOHN COTTINGHAM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100